[Description](#_Toc522474217)

[Intended User](#_Toc522474218)

[Features](#_Toc522474219)

[User Interface Mocks](#_Toc522474220)

[Key Considerations](#_Toc522474221)

[How will your app handle data persistence?](#_Toc522474222)

[Describe any edge or corner cases in the UX.](#_Toc522474223)

[Describe any libraries you’ll be using and share your reasoning for including them.](#_Toc522474224)

[Describe how you will implement Google Play Services or other external services.](#_Toc522474225)

[Next Steps: Required Tasks](#_Toc522474226)

[Task 1: Project Setup](#_Toc522474227)

[Task 2: Implement UI for Each Activity and Fragment](#_Toc522474228)

[Task 3: Setup error handling](#_Toc522474229)

[Task 4: Setup Firebase Authentication](#_Toc522474230)

[Task 5: Setup Firebase Realtime Database](#_Toc522474231)

[Task 6: Create favorite widget](#_Toc522474232)

**GitHub Username**: MatejVukosav

Personal Assistant

# Description

This app helps the user to manage his life events and information into a single app without of need to remember everything. The app solves the problem of too much information and staff that the user will use in the future but needs to remember now. For example, who will pay next drink when you see you colleague every couple of months, important notes you would like to write or what is your trousers size number in each store as they intend to have different measurements.

# Intended User

This app is intended for the regular everyday person

# Features

List the main features of your app.

* Saves personal information
* Sync among multiple devices
* Protected by user credentials

# Language

This project will be written in Java programming language.

# Others

App keeps all strings in a **string.xml** file and enables **RTL** layout switching on all layouts.

App will include support for **accessibility** through content descriptions.

**Intent service** will be implemented for widget update.

# Versions

App will be developed using following versions:

* Android Studio 3.1.3
* Support library versions: 27.1.1
* Gradle wrapper 4.5
* Gradle 3.1.4
* Constraint layout: 1.1.2
* Firebase Realtime Database: 16.0.1
* Firebase Auhentication: 16.0.3
* Firebase Crash Reporting: 16.0.1
* Glide: 4.8.0

# User Interface Mocks

|  |  |  |
| --- | --- | --- |
| Screen name | UI | Description |
| Login screen |  | The user can log in into the app using Google credentials, so his data will remain saved. This enables syncing among different devices for the same user. |
| Home screen |  | The home screen shows the main categories for the user.  Some of the categories are:   * Drinks * Notes * Clothes |
| Drinks screen |  | Drinks category show a list of people that user was with to the drink, He can see when was the last time they went together and click on a single item to see details |
| Notes screen |  | Notes category shows card items of user-created notes. The user can write as many notes as he wants. Click on single card opens card details. |
| Favorites screen |  | Favorite screen shows last viewed screens. App considered that user likes the most last viewed items and shows them in a list for easier access.  App widget will also show the same data. |
| Add new note |  | Note details enable the user to see details of the existing note and enable to change or delete the content. The user can also add new note through the details form. |
| Add new drink |  | Drink details enable the user to see details of the existing drink entry and enable to change or delete the content. The user can also add new drink entry through the details form. |
| Settings |  | In settings the user can fetch fresh data from the server, see about of the application and log out.  When a user logout all his app data are deleted from the device. |
| Widget |  | Widget will show last opened items, so user can quickly open last viewed category with a click on an item. |

# Key Considerations

### How will your app handle data persistence?

The app will save data in Firebase Realtime Database for easier synchronization with devices.

### Describe any edge or corner cases in the UX.

The user will use bottom bar to navigate between major parts of the applications. On every major screen, he can perform some actions that will lead him to new activity window.

### Describe any libraries you’ll be using and share your reasoning for including them.

Firebase – for real-time database, authentication and crashes

Glide – for image loading and caching

### Describe how you will implement Google Play Services or other external services.

I will use Firebase Realtime Database and Firebase Authentication

# Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and break them down into tangible technical tasks that you can complete one at a time until you have a finished app.

## Task 1: Project Setup

* Create an empty project.
* Configure librarires
* Enable signed build

## Task 2: Implement UI for Each Activity and Fragment

* Build UI for LoginActivity
* Build UI for MainActivity
* Build UI for HomeFragment
* Build UI for FavoritesFragment
* Build UI for SettingsFragment
* Build UI for DrinksActivity category
* Build UI for DrinksActivity details
* Build UI for NotesActivity category
* Build UI for NotesActivity details

## Task 3: Setup error handling

Setup Firebase Crash Reporting in the app so we can log user’s crashes and improve the app further.

## Task 4: Setup Firebase Authentication

Setup login into the app using user credentials. Without user credentials, the user cannot enter the app.

## Task 5: Setup Firebase Realtime Database

Enable saving the user data. When the user leaves the app, his data are persistent into Firebase database, so the user can continue on next login where he stops last time.

## Task 6: Create Favorites widget

Create UI and functionality for the favorite widget.